```
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP       SSSSSSSSSSS  YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS              YYY         YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEE                    TTT        PPP         PPP  SSS                 YYY   YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP       SSSSSSSSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUU         UUU  EEE                    TTT        PPP                       SSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP              SSSSSSSSSSS           YYY
```

FILEID**SATSSS82

```
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS    888888     222222
  SSSSSSSS    AAAAAA    TTTTTTTTTT   SSSSSSSS   SSSSSSSS   SSSSSSSS    888888     222222
SS           AA      AA      TT     SS         SS         SS           88      88  22      22
SS           AA      AA      TT     SS         SS         SS           88      88  22      22
SS           AA      AA      TT     SS         SS         SS           88      88          22
SS           AA      AA      TT     SS         SS         SS           88      88          22
  SSSSSS     AA      AA      TT       SSSSSS     SSSSSS     SSSSSS       888888          22
  SSSSSS     AA      AA      TT       SSSSSS     SSSSSS     SSSSSS       888888          22
        SS   AAAAAAAAAA      TT             SS         SS         SS   88      88      22
        SS   AAAAAAAAAA      TT             SS         SS         SS   88      88      22
        SS   AA      AA      TT             SS         SS         SS   88      88    22              ....
        SS   AA      AA      TT             SS         SS         SS   88      88    22              ....
SSSSSSSS     AA      AA      TT     SSSSSSSS   SSSSSSSS   SSSSSSSS     888888    2222222222      ....
SSSSSSSS     AA      AA      TT     SSSSSSSS   SSSSSSSS   SSSSSSSS     888888    2222222222      ....

LL             IIIIII      SSSSSSSS
LL             IIIIII      SSSSSSSS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II      SS
LL               II        SSSSSS
LL               II        SSSSSS
LL               II              SS
LL               II              SS
LL               II              SS
LL               II              SS
LLLLLLLLLL     IIIIII    SSSSSSSS
LLLLLLLLLL     IIIIII    SSSSSSSS
```

Table of contents

```
0000     1          .TITLE  SATSSS82 SATS SYSTEM SERVICE TESTS  $SETPRT (SUCC S.C.)
0000     2          .IDENT  'V04-000'
0000     3
0000     4  ;
0000     5  ;*****************************************************************************
0000     6  ;*                                                                           *
0000     7  ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     8  ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     9  ;*  ALL RIGHTS RESERVED.                                                     *
0000    10  ;*                                                                           *
0000    11  ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
0000    12  ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0000    13  ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
0000    14  ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
0000    15  ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
0000    16  ;*  TRANSFERRED.                                                             *
0000    17  ;*                                                                           *
0000    18  ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
0000    19  ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0000    20  ;*  CORPORATION.                                                             *
0000    21  ;*                                                                           *
0000    22  ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
0000    23  ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    24  ;*                                                                           *
0000    25  ;*                                                                           *
0000    26  ;*****************************************************************************
0000    27  ;
0000    28
0000    29  ;++
0000    30  ; FACILITY:     SYSTST (SATS SYSTEM SERVICE TESTS)
0000    31  ;
0000    32  ; ABSTRACT:
0000    33  ;
0000    34  ;                THIS MODULE CONTAINS SUBROUTINES WHICH, WHEN LINKED
0000    35  ; WITH SUCCOMMON.OBJ, FORM TEST MODULE SATSSS82 TO TEST SUCCESSFUL
0000    36  ; OPERATION OF THE $SETPRT SYSTEM SERVICE. THE SERVICE IS INVOKED
0000    37  ; UNDER VARIOUS INPUT CONDITIONS WITH VARYING INPUT PARAMETERS. ONLY
0000    38  ; SUCCESSFUL STATUS CODES ARE EXPECTED IN THIS TEST MODULE.  CORRECT
0000    39  ; OPERATION OF THE SERVICE FOR EACH OF ITS ISSUANCES IS VERIFIED BY
0000    40  ; CHECKING FOR AN SS$_NORMAL STATUS CODE, EXPECTED RETURN ARGUMENTS
0000    41  ; AND EXPECTED FUNCTIONALITY PERFORMED.
0000    42  ;
0000    43  ; ENVIRONMENT:  USER MODE IMAGE; NEEDS CMKRNL PRIVILEGE,
0000    44  ;                DYNAMICALLY ACQUIRES OTHER PRIVILEGES, AS NEEDED.
0000    45  ;
0000    46  ; AUTHOR: THOMAS L. CAFARELLA,          CREATION DATE: JUL, 1977
0000    47  ;
0000    48  ; MODIFIED BY:
0000    49  ;
0000    50  ;       V03-001 KDM0002         Kathleen D. Morse       28-Jun-1982
0000    51  ;                Added $PRTDEF and $SSDEF.
0000    52  ;
0000    53  ; 01    -
0000    54  ;--
```

```
                                                                I 9
0000     56          .SBTTL  DECLARATIONS
0000     57 ;
0000     58 ; INCLUDE FILES:
0000     59 ;
0000     60          $PRVDEF                         ; PRIVILEGE BIT DEFINITIONS
0000     61          $PHDDEF                         ; PROCESS HEADER OFFSETS
0000     62          $PSLDEF                         ; PROCESSOR STATUS LONGWORD DEFINITIONS
0000     63          $PRTDEF                         ; PROTECTION FIELD DEFINITIONS
0000     64          $SSDEF                          ; SYSTEM STATUS CODE DEFINITIONS
0000     65 ;
0000     66 ; MACROS:
0000     67 ;
0000     68 ;
0000     69 ; EQUATED SYMBOLS:
0000     70 ;
0000     71 ;
0000     72 ; OWN STORAGE:
0000     73 ;
```

```
                    00000000     75          .PSECT  RODATA,RD,NOWRT,NOEXE,LONG
                        0000     76 TEST_MOD_NAME:: STRING  C,<SATSSS82>     ; TEST MODULE NAME
                        0009     77 TEST_MOD_NAME_D: STRING I,<SATSSS82>     ; TEST MODULE NAME DESCRIPTOR
                        0019     78 MSG1_INP_CTL:   STRING  I,<  SSSPT!4ZW: CONDITIONS:>
                        0039     79                                          ; FAO CTL STRING FOR MSG1 IN SUCCOMMON.MAR
                        0039     80 MSG3_ERR_CTL::  STRING  I,< *SSSPT!4ZW: !AS>
                        0051     81                                          ; FAO CTL STRING FOR MSG3 IN SUCCOMMON.MAR
 04 08 05 02            0051     82 RWMODES:        .BYTE   PRT$C_KW, -      ; READ/WRITE
                        0055     83                         PRT$C_EW, -      ; .. PROT CODE
                        0055     84                         PRT$C_SW, -      ; .... FOR EACH
                        0055     85                         PRT$C_UW         ; ...... ACCESS MODE
```

```
                    87          .PSECT  RWDATA,RD,WRT,NOEXE,LONG
00000008  0000      88 PRIVMASK:        .BLKQ   1               ; ADDR OF PRIVILEGE MASK (IN PHD)
0000000C  0008      89 PROT:            .BLKL   1               ; PROT ARGUMENT FOR SETPRT
00000014  000C      90 RETADR:          .BLKQ   1               ; RETADR ARGUMENT FOR SETPRT
0000001C  0014      91 INADR:           .BLKQ   1               ; INADR ARGUMENT FOR SETPRT
0000001D  001C      92 PRVPRT:          .BLKB   1               ; PRVPRT ARGUMENT FOR SETPRT
00000025  001D      93 INADR2:          .BLKQ   1               ; INADR ARGUMENT FOR NON-SUBJECT SETPRT
```

```
                        0025     95          .SBTTL  CONDITION TABLES
                        0025     96 ;
                        0025     97 ;        ***** CONDITION TABLES FOR SETPRT SYSTEM SERVICE *****
                        0025     98 ;
                        0025     99          COND    1,NOTARG,<REGION>,-
                        0025    100                    <PROGRAM>,-
                        0025    101                    <CONTROL>,-
                        0025    102
                00000000 0045   103                        .LONG       0       ; PROGRAM
                00000001 0049   104                        .LONG       1       ; CONTROL
                        004D    105 ;
                        004D    106          COND    2,LONG,<ACMODE>,-
                        004D    107                    <KERNEL>,-
                        004D    108                    <EXEC>,-
                        004D    109                    <SUPER>,-
                        004D    110                    <USER>,-
                        004D    111
                00000000 007C   112                        .LONG       PSL$C_KERNEL
                00000001 0080   113                        .LONG       PSL$C_EXEC
                00000002 0084   114                        .LONG       PSL$C_SUPER
                00000003 0088   115                        .LONG       PSL$C_USER
                        008C    116 ;
                        008C    117          COND    3,NOTARG,<HIGH-ORDER 2 BITS OF PROT CODE>,-
                        008C    118                    <NA, RESERVED, KW, OR KR>,-
                        008C    119                    <UW, EW, ERKW, OR ER>,-
                        008C    120                    <SW, SREW, SRKW, OR SR>,-
                        008C    121                    <URSW, UREW, URKW, OR UR>,-
                        008C    122
          03 02 01 00   0116    123                        .BYTE       0,1,2,3
                        011A    124 ;
                        011A    125          COND    4,NOTARG,<LOW-ORDER 2 BITS OF PROT CODE>,-
                        011A    126                    <NA, UW, SW, OR URSW>,-
                        011A    127                    <RESERVED, EW, SREW, OR UREW>,-
                        011A    128                    <KW, ERKW, SRKW, OR URKW>,-
                        011A    129                    <KR, ER, SR, OR UR>,-
                        011A    130
          03 02 01 00   01A3    131                        .BYTE       0,1,2,3
                        01A7    132 ;
                        01A7    133          COND    5,NULL
                        01A8    134
                00000000 0000   135          .PSECT  SATSSS82,RD,WRT,EXE
```

```
                                        .SBTTL  TM_SETUP, TM_CLEANUP
                                ;++
                                ; FUNCTIONAL DESCRIPTION:
                                ;
                                ;               TM_SETUP AND TM_CLEANUP ARE CALLED TO PERFORM
                                ; REQUIRED HOUSEKEEPING AT THE BEGINNING AND END, RESPECTIVELY, OF
                                ; TEST MODULE EXECUTION.
                                ;
                                ; CALLING SEQUENCE:
                                ;
                                ;       BSBW TM_SETUP   BSBW TM_CLEANUP
                                ;
                                ; INPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT INPUTS:
                                ;
                                ;       NONE
                                ;
                                ; OUTPUT PARAMETERS:
                                ;
                                ;       NONE
                                ;
                                ; IMPLICIT OUTPUTS:
                                ;
                                ;       TM_SETUP:  COND TABLE INDEX REGISTERS (R2,3,4,5,6) CLEARED;
                                ;                  ALL PRIVILEGES ACQUIRED.
                                ;
                                ; COMPLETION CODES:
                                ;
                                ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                                ;
                                ; SIDE EFFECTS:
                                ;
                                ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                                ;       (VIA RSB) IF ERROR ENCOUNTERED.
                                ;
                                ;--
                                
                                
                                TM_SETUP::
                          52    D4    0000                CLRL    R2                      ; INITIALIZE
                          53    D4    0002                CLRL    R3                      ; .. CONDITION
                          54    D4    0004                CLRL    R4                      ; .... TABLE
                          55    D4    0006                CLRL    R5                      ; ...... INDEX
                          56    D4    0008                CLRL    R6                      ; ........ REGISTERS
                      FFF3'     30    000A                BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE BEGIN MSG
00000000'EF   00000000'EF     DE    000D                MOVAL   TEST_MOD_SUCC,TMD_ADDR  ; ASSUME END MSG WILL SHOW SUCCESS
   03   00   00000000'8F     F0    0018                INSV    #SUCCESS,#0,#3,MOD_MSG_CODE ; ADJUST STATUS CODE FOR SUCCESS
             00000000'EF           0020
                                   0025                MODE    TO,5$,KRNL              ; KERNEL MODE TO ACCESS PHD
         59  00000000'9F     D0    0048                MOVL    @#CTL$GL_PHD,R9         ; GET PROCESS HEADER ADDRESS
         00000000'EF   69    DE    004F                MOVAL   PHD$Q_PRIVMSK(R9),PRIVMASK ; GET PRIV MASK ADDRESS
                               0056                     MODE    FROM,5$ ; BACK TO USER MODE
                               0057                     PRIV    ADD,ALL                 ; GET ALL PRIVILEGES
```

```
                 0077   193          $SETPRN_S TEST_MOD_NAME_D       ; SET PROCESS NAME
                 0084   194          SS_CHECK  NORMAL                ; CHECK STATUS CODE RETURNED FROM SETPRN
           05    00AE   195          RSB                             ; RETURN TO MAIN ROUTINE
                 00AF   196 TM_CLEANUP::
FF4E'      30    00AF   197          BSBW    MOD_MSG_PRINT           ; PRINT TEST MODULE END MSG
           05    00B2   198          RSB                             ; RETURN TO MAIN ROUTINE
```

```
                                                B 10
00B3    200          .SBTTL  CONDITION SUBROUTINES - SETUP AND CLEANUP
00B3    201 ;++
00B3    202 ; FUNCTIONAL DESCRIPTION:
00B3    203 ;
00B3    204 ;               CONDX AND CONDX_CLEANUP ARE SUBROUTINES WHICH ARE EXECUTED
00B3    205 ; BEFORE AND AFTER THE VERIFY SUBROUTINE, RESPECTIVELY, WHENEVER A NEW
00B3    206 ; CONDITION X VALUE IS SELECTED (SEE FUNCTIONAL DESCRIPTION OF SUCCOMMON
00B3    207 ; ROUTINE IN SUCCOMMON.MAR). ANY SETUP FUNCTION PARTICULAR TO THE
00B3    208 ; CONDITION X TABLE IS INCLUDED IN THE CONDX SUBROUTINE AND CLEANED
00B3    209 ; UP, IF NECESSARY, IN THE CONDX_CLEANUP SUBROUTINE. THIS INCLUDES,
00B3    210 ; ESPECIALLY, CODE TO DETECT CONFLICTS AMONG CURRENT ENTRIES IN TWO
00B3    211 ; OR MORE CONDITION TABLES. IF A CONFLICT IS DETECTED, A NON-ZERO
00B3    212 ; VALUE IS STORED INTO CONFLICT, WHICH CAUSES THE CALLING ROUTINE
00B3    213 ; (SUCCOMMON) TO SKIP THE CURRENT ENTRY IN THE CONDITION X TABLE.
00B3    214 ;
00B3    215 ; CALLING SEQUENCE:
00B3    216 ;
00B3    217 ;       BSBW CONDX   BSBW CONDX_CLEANUP
00B3    218 ;         WHERE X = 1,2,3,4,5
00B3    219 ;
00B3    220 ; INPUT PARAMETERS:
00B3    221 ;
00B3    222 ;       CONFLICT = 0
00B3    223 ;
0083    224 ; IMPLICIT INPUTS:
0083    225 ;
0083    226 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
0083    227 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
0083    228 ;
0083    229 ; OUTPUT PARAMETERS:
0083    230 ;
0083    231 ;       CONFLICT SET TO NON-ZERO IF COND TABLE CONFLICT DETECTED.
0083    232 ;
0083    233 ; IMPLICIT OUTPUTS:
0083    234 ;
0083    235 ;       R2,3,4,5,6 PRESERVED
0083    236 ;
0083    237 ; COMPLETION CODES:
0083    238 ;
0083    239 ;       NONE
0083    240 ;
0083    241 ; SIDE EFFECTS:
0083    242 ;
0083    243 ;       NONE
0083    244 ;
0083    245 ;--
0083    246
0083    247
0083    248
0083    249 COND1::
05  0083    250         RSB                             ; RETURN TO MAIN ROUTINE
0084    251 COND1_CLEANUP::
05  0084    252         RSB                             ; RETURN TO MAIN ROUTINE
0085    253 COND2::
05  0085    254         RSB                             ; RETURN TO MAIN ROUTINE
0086    255 COND2_CLEANUP::
05  0086    256         RSB                             ; RETURN TO MAIN ROUTINE
```

```
                        00B7   257 COND3::
                     05 00B7   258         RSB                                ; RETURN TO MAIN ROUTINE
                        00B8   259 COND3_CLEANUP::
                     05 00B8   260         RSB                                ; RETURN TO MAIN ROUTINE
                        00B9   261 COND4::
          00000116'EF44 95 00B9   262         TSTB    COND3_E[R4]                ; FIRST CONDITION 3 ELEMENT ?
                     15 12 00C0   263         BNEQ    COND4X                     ; NO -- ALL IS OK
       01 000001A3'EF45 91 00C2   264         CMPB    COND4_E[R5],#1             ; 2ND CONDITION 4 ELEMENT ?
                     0B 12 00CA   265         BNEQ    COND4X                     ; NO -- ALL IS OK
                        00CC   266                                                ; YES -- THIS PROT CODE RESERVED
00000000'EF   00000000'EF 90 00CC   267         MOVB    ONES,CONFLICT              ; INDICATE CONFLICT
                        00D7   268 COND4X:
                     05 00D7   269         RSB                                ; RETURN TO MAIN ROUTINE
                        00D8   270 COND4_CLEANUP::
                     05 00D8   271         RSB                                ; RETURN TO MAIN ROUTINE
                        00D9   272 COND5::
                     05 00D9   273         RSB                                ; RETURN TO MAIN ROUTINE
                        00DA   274 COND5_CLEANUP::
                     05 00DA   275         RSB                                ; RETURN TO MAIN ROUTINE
```

```
                                                D 10
00DB   277          .SBTTL  FORM_CONDS
00DB   278 ;++
00DB   279 ; FUNCTIONAL DESCRIPTION:
00DB   280 ;
00DB   281 ;                FORM_CONDS FORMATS AND PRINTS INFORMATION ABOUT
00DB   282 ;  THE CURRENT ELEMENT IN EACH OF THE CONDITION TABLES.
00DB   283 ;
00DB   284 ; CALLING SEQUENCE:
00DB   285 ;
00DB   286 ;       BSBW FORM_CONDS
00DB   287 ;
00DB   288 ; INPUT PARAMETERS:
00DB   289 ;
00DB   290 ;       NONE
00DB   291 ;
00DB   292 ; IMPLICIT INPUTS:
00DB   293 ;
00DB   294 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
00DB   295 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
00DB   296 ;       FOR X = 1,2,3,4,5 :
00DB   297 ;               CONDX_T - TITLE TEXT FOR CONDX TABLE
00DB   298 ;               CONDX_TAB - ELEMENT TEXT FOR CONDX TABLE
00DB   299 ;               CONDX_C - CONTEXT OF THE CONDX TABLE
00DB   300 ;               CONDX_E - DATA ELEMENTS OF THE CONDX TABLE
00DB   301 ;
00DB   302 ; OUTPUT PARAMETERS:
00DB   303 ;
00DB   304 ;       NONE
00DB   305 ;
00DB   306 ; IMPLICIT OUTPUTS:
00DB   307 ;
00DB   308 ;       NONE
00DB   309 ;
00DB   310 ; COMPLETION CODES:
00DB   311 ;
00DB   312 ;       NONE
00DB   313 ;
00DB   314 ; SIDE EFFECTS:
00DB   315 ;
00DB   316 ;       NONE
00DB   317 ;
00DB   318 ;--
00DB   319
00DB   320
00DB   321
00DB   322 FORM_CONDS::
00DB   323          $FAO_S  MSG1_INP_CTL,FAO_LEN,FAO_DESC,TESTNUM
00FA   324                                              ; FORMAT CONDITIONS HEADER MSG
FF03'  30  00FA   325          BSBW    OUTPUT_MSG       ;   ... AND PRINT IT
  14   00  91  00FD   326          CMPB    #COND1_C,#NULL   ; IS CONDITION 1 NULL ?
       03  12  0100   327          BNEQU   10$              ; NO -- CONTINUE
     00CB  31  0102   328          BRW     FORM_CONDSX      ; YES -- SUBROUTINE IS FINISHED
                0105   329 10$:
00000000'EF  00000025'EF  DE  0105   330          MOVAL   COND1_T,MSG_A    ; SAVE ADDRESS OF CONDITION 1 TITLE FOR FAO
00000000'EF  0000002D'EF42  D0  0110   331          MOVL    COND1_TAB[R2],MSG_B  ; SAVE ADDR OF COND 1 CURR TEXT ELT FOR FAO
        00000000'EF  00  90  011C   332          MOVB    #COND1_C,MSG_CTXT    ; SAVE CONDITION 1 CONTEXT FOR FAO
                0123   333          MOV_VAL COND1_C,COND1_E[R2],MSG_DATA1 ; GIVE COND 1 DATA VALUE TO FAO
```

```
                     FEDA'  30  0123   334          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 1 MSG
                 14    04   91  0126   335          CMPB    #COND2_C,#NULL                  ; IS CONDITION 2 NULL ?
                       03   12  0129   336          BNEQU   20$                             ; NO -- CONTINUE
                     00A2   31  012B   337          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            012E   338 20$:
  00000000'EF    0000004D'EF   DE  012E   339          MOVAL   COND2_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 2 TITLE FOR FAO
00000000'EF    00000055'EF43   D0  0139   340          MOVL    COND2_TAB[R3],MSG_B             ; SAVE ADDR OF COND 2 CURR TEXT ELT FOR FAO
          00000000'EF    04   90  0145   341          MOVB    #COND2_C,MSG_CTXT               ; SAVE CONDITION 2 CONTEXT FOR FAO
                            014C   342          MOV_VAL COND2_C,COND2_E[R3],MSG_DATA1 ; GIVE COND 2 DATA VALUE TO FAO
                     FEA5'  30  0158   343          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 2 MSG
                 14    00   91  015B   344          CMPB    #COND3_C,#NULL                  ; IS CONDITION 3 NULL ?
                       03   12  015E   345          BNEQU   30$                             ; NO -- CONTINUE
                     006D   31  0160   346          BRW     FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
                            0163   347 30$:
  00000000'EF    0000008C'EF   DE  0163   348          MOVAL   COND3_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 3 TITLE FOR FAO
00000000'EF    000000AC'EF44   D0  016E   349          MOVL    COND3_TAB[R4],MSG_B             ; SAVE ADDR OF COND 3 CURR TEXT ELT FOR FAO
          00000000'EF    00   90  017A   350          MOVB    #COND3_C,MSG_CTXT               ; SAVE CONDITION 3 CONTEXT FOR FAO
                            0181   351          MOV_VAL COND3_C,COND3_E[R4],MSG_DATA1 ; GIVE COND 3 DATA VALUE TO FAO
                     FE7C'  30  0181   352          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 3 MSG
                 14    00   91  0184   353          CMPB    #COND4_C,#NULL                  ; IS CONDITION 4 NULL ?
                       47   13  0187   354          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF    0000011A'EF   DE  0189   355          MOVAL   COND4_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 4 TITLE FOR FAO
00000000'EF    00000139'EF45   D0  0194   356          MOVL    COND4_TAB[R5],MSG_B             ; SAVE ADDR OF COND 4 CURR TEXT ELT FOR FAO
          00000000'EF    00   90  01A0   357          MOVB    #COND4_C,MSG_CTXT               ; SAVE CONDITION 4 CONTEXT FOR FAO
                            01A7   358          MOV_VAL COND4_C,COND4_E[R5],MSG_DATA1 ; GIVE COND 4 DATA VALUE TO FAO
                     FE56'  30  01A7   359          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 4 MSG
                 14    14   91  01AA   360          CMPB    #COND5_C,#NULL                  ; IS CONDITION 5 NULL ?
                       21   13  01AD   361          BEQLU   FORM_CONDSX                     ; YES -- SUBROUTINE IS FINISHED
  00000000'EF    000001A7'EF   DE  01AF   362          MOVAL   COND5_T,MSG_A                   ; SAVE ADDRESS OF CONDITION 5 TITLE FOR FAO
00000000'EF    000001A7'EF46   D0  01BA   363          MOVL    COND5_TAB[R6],MSG_B             ; SAVE ADDR OF COND 5 CURR TEXT ELT FOR FAO
          00000000'EF    14   90  01C6   364          MOVB    #COND5_C,MSG_CTXT               ; SAVE CONDITION 5 CONTEXT FOR FAO
                            01CD   365          MOV_VAL COND5_C,COND5_E[R6],MSG_DATA1 ; GIVE COND 5 DATA VALUE TO FAO
                     FE30'  30  01CD   366          BSBW    WRITE_MSG2                      ; FORMAT AND WRITE CONDITION 5 MSG
                            01D0   367 FORM_CONDSX:
                       05   01D0   368          RSB                                     ; RETURN TO CALLER
```

```
01D1    370          .SBTTL  VERIFY
01D1    371 ;++
01D1    372 ; FUNCTIONAL DESCRIPTION:
01D1    373 ;
01D1    374 ;               VERIFY IS CALLED ONCE FOR EACH COMBINATION OF CONDITION
01D1    375 ; TABLE VALUES (AS DETERMINED BY THE INDEX REGISTERS R2,3,4,5,6 FOR
01D1    376 ; COND TABLES 1,2,3,4,5, RESPECTIVELY). VERIFY ESTABLISHES THE CONDITIONS
01D1    377 ; SPECIFIED BY THE COND TABLES AND ISSUES THE SUBJECT SYSTEM SERVICE
01D1    378 ; ($SETPRT). THEN, THE SUCCESSFUL OPERATION OF THE SERVICE IS VERIFIED
01D1    379 ; BY EXAMINING THE STATUS CODE RETURNED, THE VALUES FOR RETURN ARGUMENTS
01D1    380 ; AND THE FUNCTIONALITY PERFORMED. THE EXAMINATIONS TAKE THE FORM OF
01D1    381 ; COMPARISONS AGAINST EXPECTED VALUES. ANY FAILING COMPARISON CAUSES AN
01D1    382 ; ERR_EXIT MACRO TO BE EXECUTED (EITHER DIRECTLY, OR INDIRECTLY,
01D1    383 ; THROUGH THE SS_CHECK MACRO); ERR_EXIT SETS EFLAG TO NON-ZERO,
01D1    384 ; PRINTS ERROR MESSAGES AND CAUSES AN IMMEDIATE RSB TO CALLER.
01D1    385 ; WHEN ERR_EXIT IS EXECUTED, FURTHER CALLS TO VERIFY ARE SUPPRESSED,
01D1    386 ; AND, AFTER EXECUTING CLEANUP SUBROUTINES, THE IMAGE EXITS.
01D1    387 ;
01D1    388 ; CALLING SEQUENCE:
01D1    389 ;
01D1    390 ;       BSBW VERIFY
01D1    391 ;
01D1    392 ; INPUT PARAMETERS:
01D1    393 ;
01D1    394 ;       NONE
01D1    395 ;
01D1    396 ; IMPLICIT INPUTS:
01D1    397 ;
01D1    398 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
01D1    399 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
01D1    400 ;       FOR X = 1,2,3,4,5 :
01D1    401 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
01D1    402 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
01D1    403 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
01D1    404 ;                 FOR CONDX_E.
01D1    405 ;
01D1    406 ; OUTPUT PARAMETERS:
01D1    407 ;
01D1    408 ;       NONE
01D1    409 ;
01D1    410 ; IMPLICIT OUTPUTS:
01D1    411 ;
01D1    412 ;       VERIFY HAS NO OUTPUT. SINCE ITS PURPOSE IS TO TEST FOR ERRORS,
01D1    413 ;       IT MERELY RETURNS TO CALLER NORMALLY AFTER THE TESTS, PROVIDING
01D1    414 ;       ALL WERE SUCCESSFUL; IF AN ERROR IS DISCOVERED, RETURN IS VIA
01D1    415 ;       AN ERR_EXIT OR SS_CHECK MACRO, BOTH OF WHICH DOCUMENT DETECTED
01D1    416 ;       ERRORS.
01D1    417 ;
01D1    418 ; COMPLETION CODES:
01D1    419 ;
01D1    420 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
01D1    421 ;
01D1    422 ; SIDE EFFECTS:
01D1    423 ;
01D1    424 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
01D1    425 ;       (VIA RSB) IF ERROR ENCOUNTERED.
01D1    426 ;
```

```
                                01D1   427 ;--
                                01D1   428
                                01D1   429
                                01D1   430
                                01D1   431 VERIFY::
                00000000'EF   95 01D1   432         TSTB    CFLAG                   ; SHOULD CONDITIONS BE PRINTED ?
                          03   13 01D7   433         BEQL    5$                      ; NO -- CONTINUE
                        FEFF   30 01D9   434         BSBW    FORM_CONDS              ; YES -- FMT & PRINT ALL CONDS FOR THIS T.C.
                                01DC   435 5$:
                00000014'EF   7C 01DC   436         CLRQ    INADR                   ; CLEAR TO INDICATE EXPREG NOT ISSUED YET
                                01E2   437         MODE    TO,7$,KRNL              ; LET ACMODE ARGUMENT CONTROL MODE
                                0205   438         $EXPREG_S PAGCNT=#25, RETADR=INADR, ACMODE=ACMODE[R3], -
                                0205   439                   REGION=COND1_E[R2]    ; GRAB A HUNK OF SPACE
                                0222   440         MODE    FROM,7$                 ; BACK TO USER MODE
                                0223   441         SS_CHECK NORMAL                 ; CHECK NORMAL COMPLETION
              00000008'EF    FF 8F   98 024D   442         CVTBL   #-1,PROT                ; TRY TO CONFUSE SETPRT WITH HI-ORDER BITS
00000008'EF    000001A3'EF45   90 0255   443         MOVB    COND4_E[R5],PROT        ; GET 2 LOW-ORDER BITS OF PROT CODE
          59    00000116'EF44   90 0261   444         MOVB    COND3_E[R4],R9          ; PICK UP 2 HI-ORDER BITS ...
 00000008'EF    02    02   59   F0 0269   445         INSV    R9,#2,#2,PROT           ; ... AND FORM A COMPLETE CODE
                                0272   446         MODE    TO,10$,KRNL             ; LET ACMODE ARGUMENT CONTROL MODE
                                0295   447 ;
                                0295   448 ; ****** SYSTEM SERVICE CALL WHICH IS THE SUBJECT OF THIS TEST CASE ******
                                0295   449 ;
                                0295   450         $SETPRT_S INADR,RETADR,ACMODE[R3],PROT,PRVPRT
                                02BB   451         MODE    FROM,10$                ; BACK TO USER MODE
                          01    50   D1 02BC   452         CMPL    R0,#SS$_NORMAL          ; CODE RECEIVED = CODE EXPECTED ?
                                03   12 02BF   453         BNEQU   20$                     ; NO -- GO PROCESS ERROR
                              005D   31 02C1   454         BRW     30$                     ; YES -- CONTINUE
                                02C4   455 20$:
                00000000'EF   01   D0 02C4   456         MOVL    #SS$_NORMAL,EXPV        ; LOAD UP EXPECTED AND ...
                00000000'EF   50   D0 02CB   457         MOVL    R0,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                                02D2   458         ERR_EXIT LONG,<INCORRECT STATUS CODE RETURNED FROM SETPRT>
                                0321   459 30$:
  0000000C'EF   00000014'EF   D1 0321   460         CMPL    INADR,RETADR            ; DID SETPRT BEGIN AT RIGHT PLACE ?
                                0F   12 032C   461         BNEQU   35$                     ; NO -- ERROR
  00000010'EF   00000018'EF   D1 032E   462         CMPL    INADR+4,RETADR+4        ; DID IT DO ALL OF THE PAGES ?
                                02   12 0339   463         BNEQU   35$                     ; NO -- THAT'S NO GOOD EITHER
                                6E   11 033B   464         BRB     40$                     ; YES -- ALL IS OK, SO FAR
                                033D   465 35$:
  00000000'EF   00000014'EF   7D 033D   466         MOVQ    INADR,EXPV              ; LOAD UP PAIR OF EXPECTED ...
  00000000'EF   0000000C'EF   7D 0348   467         MOVQ    RETADR,RECV             ; ... AND RECEIVED VALUES, THEN EXIT
                                0353   468         ERR_EXIT QUAD,<SETPRT DID NOT FUNCTION >, -
                                0353   469                       <ON REQUESTED RANGE OF PAGES>
                                03AB   470 40$:
00000051'EF43   0000001C'EF   91 03AB   471         CMPB    PRVPRT,RWMODES[R3]      ; WAS PREV PROT R/W ?
                                65   13 03B7   472         BEQLU   60$                     ; YES -- DO MORE CHECKING
00000000'EF    00000051'EF43   90 03B9   473         MOVB    RWMODES[R3],EXPV        ; NO -- LOAD UP EXPECTED AND ...
  00000000'EF   0000001C'EF   90 03C5   474         MOVB    PRVPRT,RECV             ; ... RECEIVED VALUES, THEN EXIT
                                03D0   475         ERR_EXIT BYTE,<INCORRECT PRVPRT VALUE RETURNED BY SETPRT>
                                041E   476 60$:
                00   00000008'EF   91 041E   477         CMPB    PROT,#PRT$C_NA          ; IS PROT CODE NO-ACCESS ?
                                07   12 0425   478         BNEQU   70$                     ; NO -- CONTINUE
                00000008'EF   03   90 0427   479         MOVB    #PRT$C_KR,PROT          ; YES -- NA IS CHANGED TO KR
                                042E   480 70$:
  0000001D'EF   00000014'EF   D0 042E   481         MOVL    INADR,INADR2            ; REQUEST A SETPRT ON 1 PAGE ...
  00000021'EF   00000014'EF   D0 0439   482         MOVL    INADR,INADR2+4          ; ... TO VERIFY PROT CODE
                                0444   483         MODE    TO,65$,KRNL             ; GET KERNEL MODE
```

```
                             0467   484          $SETPRT_S INADR2,RETADR,ACMODE[R3],PROT,PRVPRT
                             048D   485          MODE    FROM,65$                    ; BACK TO USER MODE
                             048E   486          SS_CHECK NORMAL                     ; CHECK FOR NORMAL RETURN
00000008'EF   0000001C'EF 91 04B8   487          CMPB    PRVPRT,PROT                 ; DID SETPRT RETURN SAME PROT VALUE ?
                        5C 13 04C3   488          BEQLU   VERIFYX                     ; YES -- OK
00000000'EF   00000008'EF 90 04C5   489          MOVB    PROT,EXPV                   ; NO -- LOAD UP EXPECTED AND ...
00000000'EF   0000001C'EF 90 04D0   490          MOVB    PRVPRT,RECV                 ; ... RECEIVED VALUES, THEN EXIT
                             04DB   491          ERR_EXIT BYTE,<PROT CODE NOT PRESERVED BY SETPRT>
                             0521   492 VERIFYX:
                          05 0521   493          RSB                                 ; RETURN TO CALLER
```

```
                     0522    495          .SBTTL  VFY_CLEANUP
                     0522    496 ;++
                     0522    497 ; FUNCTIONAL DESCRIPTION:
                     0522    498 ;
                     0522    499 ;               VFY_CLEANUP EXECUTES SYSTEM SERVICES TO UNDO THE
                     0522    500 ; EFFECT OF THOSE ISSUED IN THE VERIFY SUBROUTINE. VFY_CLEANUP MUST
                     0522    501 ; ASSUME THAT VERIFY MAY NOT HAVE EXECUTED IN ITS ENTIRETY (IF AN
                     0522    502 ; ERROR IS FOUND). ALSO, VFY_CLEANUP MAY ISSUE SS_CHECK OR ERR_EXIT
                     0522    503 ; ONLY AFTER PERFORMING ALL OF ITS CLEANUP OPERATIONS; THIS IS REQUIRED
                     0522    504 ; IN THE EVENT THAT VFY_CLEANUP IS CALLED DURING ERROR PROCESSING,
                     0522    505 ; WHEN PERFORMING THE REQUIRED CLEANUP IS MORE IMPORTANT THAN
                     0522    506 ; POSSIBLY DISCOVERING A SECOND ERROR.
                     0522    507 ;
                     0522    508 ; CALLING SEQUENCE:
                     0522    509 ;
                     0522    510 ;       BSBW VFY_CLEANUP
                     0522    511 ;
                     0522    512 ; INPUT PARAMETERS:
                     0522    513 ;
                     0522    514 ;       NONE
                     0522    515 ;
                     0522    516 ; IMPLICIT INPUTS:
                     0522    517 ;
                     0522    518 ;       R2,3,4,5,6 CONTAIN CURRENT CONDITION TABLE INDEX VALUES
                     0522    519 ;         FOR COND TABLES 1,2,3,4,5, RESPECTIVELY.
                     0522    520 ;       FOR X = 1,2,3,4,5 :
                     0522    521 ;               CONDX_E - ADDRESS OF TABLE OF DATA VALUES FOR CONDX
                     0522    522 ;                 TABLE. IF THE CONTEXT OF TABLE X IS A SYSTEM SERVICE
                     0522    523 ;                 ARGUMENT, THE ARGUMENT NAME MAY BE USED AS A SYNONYM
                     0522    524 ;                 FOR CONDX_E.
                     0522    525 ;
                     0522    526 ; OUTPUT PARAMETERS:
                     0522    527 ;
                     0522    528 ;       NONE
                     0522    529 ;
                     0522    530 ; IMPLICIT OUTPUTS:
                     0522    531 ;
                     0522    532 ;       NONE
                     0522    533 ;
                     0522    534 ; COMPLETION CODES:
                     0522    535 ;
                     0522    536 ;       EFLAG SET TO NON-ZERO IF ERROR ENCOUNTERED.
                     0522    537 ;
                     0522    538 ; SIDE EFFECTS:
                     0522    539 ;
                     0522    540 ;       SS_CHECK AND ERR_EXIT MACROS CAUSE PREMATURE EXIT
                     0522    541 ;       (VIA RSB) IF ERROR ENCOUNTERED.
                     0522    542 ;
                     0522    543 ;--
                     0522    544
                     0522    545
                     0522    546
                     0522    547 VFY_CLEANUP::
00000014'EF   D5     0522    548          TSTL    INADR                  ; DID EXPREG GET ISSUED SUCCESSFULLY ?
          03   12    0528    549          BNEQ    10$                    ; YES -- CONTINUE
        0064   31    052A    550          BRW     VFY_CLEANUPX           ; NO -- JUST EXIT
                     052D    551 10$:
```

```
                                                J 10
     052D   552          MODE    TO,20$,KRNL                ; NEED KERNEL TO SPECIFY MODE
     0550   553          $DELTVA_S INADR,,ACMODE[R3]         ; GET RID OF EXPANDED REGION
     0566   554          MODE    FROM,20$                   ; BACK TO USER MODE
     0567   555          SS_CHECK NORMAL                    ; CHECK RETURN FROM DELTVA
     0591   556 VFY_CLEANUPX:
05   0591   557          RSB                                ; RETURN TO CALLER
     0592   558          .END
```

```
$$$$                    = 000004E5 R     04
$$$CHARS                = 00000021
$$$CHARS1               = 00000013
$$$CHARS2               = 0000001B
$$$CHARS3               = 00000017
$$$CHARS4               = 00000011
$$$CHARS5               = 00000000
$$$COND_A               = 00000003
$$$STRINGS              = 00000001
$$$STRINGS2             = 00000005
$$T1                    = 00000000
$$T2                    = 00000004
ACMODE                    0000007C R     03
BYTE                    = 00000001   G
CFLAG                     ********   X   04
CHMRTN                    ********   X   04
CHM_CONT                  ********   X   04
COMP_SC                   ********   X   04
CONDT                     000000B3 RG    04
COND1_C                 = 00000000
COND1_CLEANUP             000000B4 RG    04
COND1_E                   00000045 R     03
COND1_H                   0000002C RG    03
COND1_T                   00000025 R     03
COND1_TAB                 0000002D R     03
COND2                     000000B5 RG    04
COND2_C                 = 00000004
COND2_CLEANUP             000000B6 RG    04
COND2_E                   0000007C R     03
COND2_H                   00000054 RG    03
COND2_T                   0000004D R     03
COND2_TAB                 00000055 R     03
COND3                     000000B7 RG    04
COND3_C                 = 00000000
COND3_CLEANUP             000000B8 RG    04
COND3_E                   00000116 R     03
COND3_H                   000000AB RG    03
COND3_T                   0000008C R     03
COND3_TAB                 000000AC R     03
COND4                     000000B9 RG    04
COND4X                    000000D7 R     04
COND4_C                 = 00000000
COND4_CLEANUP             000000D8 RG    04
COND4_E                   000001A3 R     03
COND4_H                   00000138 RG    03
COND4_T                   0000011A R     03
COND4_TAB                 00000139 R     03
COND5                     000000D9 RG    04
COND5_C                 = 00000014
COND5_CLEANUP             000000DA RG    04
COND5_H                   000001A7 RG    03
COND5_T                   000001A7 R     03
COND5_TAB                 000001A7 R     03
CONFLICT                  ********     X 04
CTL$GL_PHD                ********     X 04
DESC                    = 00000010   G
EFLAG                     ********   X   04
EXPV                      ********   X   04
FAO_DESC                  ********   X   04
FAO_LEN                   ********   X   04
FORM_CONDS                000000DB RG    04
FORM_CONDSX               000001D0 R     04
INADR                     00000014 R     03
INADR2                    0000001D R     03
LONG                    = 00000004   G
MOD_MSG_CODE              ********   X   04
MOD_MSG_PRINT             ********   X   04
MSGT_INP_CTL              00000019 R     02
MSG3_ERR_CTL              00000039 RG    02
MSG_A                     ********   X   04
MSG_B                     ********   X   04
MSG_CTXT                  ********   X   04
MSG_DATA1                 ********   X   04
NOTARG                  = 00000000   G
NULL                    = 00000014   G
ONES                      ********   X   04
OUTPUT_MSG                ********   X   04
PCV                       ********   X   04
PHD$Q_PRIVMSK           = 00000000
PRIVMASK                  00000000 R     03
PRIV_ARGS               = 00000002
PROCESS_ERR               ********   X   04
PROT                      0000000B R     03
PRT$C_EW                = 00000005
PRT$C_KR                = 00000003
PRT$C_KW                = 00000002
PRT$C_NA                = 00000000
PRT$C_SW                = 00000008
PRT$C_UW                = 00000004
PRVPRT                    0000001C R     03
PSL$C_EXEC              = 00000001
PSL$C_KERNEL            = 00000000
PSL$C_SUPER             = 00000002
PSL$C_USER              = 00000003
QUAD                    = 00000008   G
RECV                      ********   X   04
REST_REGS                 ********   X   04
RETADR                    0000000C R     03
RWMODES                   00000051 R     02
SAVE_REGS                 ********   X   04
SS$_NORMAL              = 00000001
SUCCESS                   ********   X   04
SYS$CMKRNL                ********  GX   04
SYS$DELTVA                ********  GX   04
SYS$EXPREG                ********  GX   04
SYS$FAO                   ********   X   04
SYS$SETPRN                ********  GX   04
SYS$SETPRT                ********  GX   04
SYS$SETPRV                ********  GX   04
TESTNUM                   ********   X   04
TEST_MOD_NAME             00000000 RG    02
TEST_MOD_NAME_D           00000009 R     02
TEST_MOD_SUCC             ********   X   04
TMD_ADDR                  ********   X   04
```

```
TM_CLEANUP                      000000AF RG    04
TM_SETUP                        00000000 RG    04
VERIFY                          000001D1 RG    04
VERIFYX                         00000521 R     04
VFY_CLEANUP                     00000522 RG    04
VFY_CLEANUPX                    00000591 R     04
WORD                          = 00000002   G
WRITE_MSG2                      ********     X 04
```

+----------------+
! Psect synopsis !
+----------------+

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
RODATA	00000055 (85.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
RWDATA	000001A8 (424.)	03 (3.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC LONG
SATSSS82	00000592 (1426.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

+------------------------+
! Performance indicators !
+------------------------+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.06	00:00:00.56
Command processing	107	00:00:00.74	00:00:01.92
Pass 1	329	00:00:10.97	00:00:16.85
Symbol table sort	0	00:00:01.25	00:00:01.41
Pass 2	122	00:00:02.41	00:00:03.12
Symbol table output	15	00:00:00.10	00:00:00.13
Psect synopsis output	2	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	606	00:00:15.57	00:00:24.03

```
The working set limit was 1350 pages.
59058 bytes (116 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 767 non-local and 47 local symbols.
558 source lines were read in Pass 1, producing 24 object records in Pass 2.
39 pages of virtual memory were used to define 30 macros.
```

+---------------------------+
! Macro library statistics !
+---------------------------+

Macro library name	Macros defined
_255DUA28:[SHRLIB]UETP.MLB;1	9
_255DUA28:[SYS.OBJ]LIB.MLB;1	1
_255DUA28:[SYSLIB]STARLET.MLB;2	17
TOTALS (all libraries)	27

```
1113 GETS were required to define 27 macros.

There were no errors, warnings or information messages.
```

```
MACRO/LIS=LIS$:SATSSS82/OBJ=OBJ$:SATSSS82 MSRC$:SATSSS82/UPDATE=(ENH$:SATSSS82)+EXECML$/LIB+SHRLIB$:UETP/LIB
```

SATSS574
LIS

SATSUT01
LIS

SATSUT05
LIS

SATSS582
LIS

SATSS590
LIS

SATSS578
LIS

SATSS580
LIS

SATSUT04
LIS

SATSUT06
LIS

SATSS581
LIS

SATSS591
LIS

SATSS579
LIS

SATSS583
LIS